УДК 681.3.06

ПОСТРОЕНИЕ ОПТИМАЛЬНОГО РАСПРЕДЕЛЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ РЕСУРСОВ ПО ПРОГРАММНЫМ КОМПЛЕКСАМ ОБРАБОТКИ

М.А. Сонькин, В.В. Грачев*, А.Л. Кузьмин**, М.А. Яценко*

Томский политехнический университет
*ФГУП «НИИ «Квант», г. Москва
**Академия ФСО, г. Орел
E-mail: sonkin@tpu.ru

Рассматривается моделирование работы системы конвейерной обработки информации, позволяющее повысить эффективность ее функционирования. Построено псевдооптимальное распределение программных комплексов обработки по вычислительным модулям системы конвейерной обработки информации.

Ключевые слова:
Системы конвейерной обработки информации, псевдооптимальное распределение программных комплексов, генетический алгоритм.

Key words:
Conveyor system of information processing, a pseudo optimal distribution of software systems, genetic algorithm.

Введение

В настоящее время системы конвейерной обработки информации (СКОИ) получили широкое распространение. В систему поступает случайный поток информационных элементов из разных источников и в различных форматах. Далее происходит их обработка в режиме многошагового аналитического конвейера, включающего стадии преобразования и интеграции данных.

Во время работы ресурсы СКОИ распределяются неравномерно, что позволяет считать, что часть производительности остается нереализованной. Моделирование работы СКОИ является одним из подходов, позволяющим повысить эффективность функционирования [1].

При проектировании СКОИ возникает задача определения количества вычислительных ресурсов необходимых для обработки заданного входного потока. Для решения этой задачи успешно используется имитационное моделирование [2]. На основании собранной статистической информации строится имитационная модель системы и при различных параметрах входного потока определяется загруженность ее компонентов.

После того, как определено количество вычислительных ресурсов, необходимых для обработки входного потока, возникает вопрос об их оптимальном распределении. Решение этой задачи позволит обрабатывать входной поток с меньшими затратами, не теряя при этом производительности. Построение оптимального распределения вычислительных ресурсов СКОИ можно интерпретировать как задачу нелинейного целочисленного программирования. Одним из методов решения сложных оптимизационных задач, в частности, с нелинейными целевыми функциями, является применение генетического алгоритма [3]. Однако ввиду сложности ограничений и целевой функции использование целочисленного генетического алгоритма неэффективно, поэтому полученное решение приводится к целочисленному псевдооптимальному с помощью преобразования, основанного на использовании «жадного» алгоритма поиска.

В последующих разделах рассматривается построение псевдооптимального распределения программных комплексов обработки (ПКО) по вычислительным модулям (ВМ) системы конвейерной обработки информации.

Система конвейерной обработки информации

Система конвейерной обработки информации принимает случайный поток информационных элементов.

Информационный элемент (ИЭ) – единица обрабатываемой информации в СКОИ.

Процесс конвейерной обработки информации P_i описывается как $P_i=\langle t_i,A_i,Tr_i\rangle$; где t_i – момент инициирования процесса; A_i – атрибуты процесса, определяющие параметры источника информации, параметры информационного элемента, режим обработки данных, приоритет процесса и др.; Tr_i – трасса процесса.

Трасса процесса – последовательность этапов обработки, связанных с изменением информационного элемента от этапа к этапу. Трасса процесса представляется в виде упорядоченного множества этапов $Tr_i=\{S_1,S_2,t_{k_i}\}$, имевших место в моменты времени t_1, t_2, t_{k_i}, такие, что $t_1\le t_2\le t_{k_i}$. К этапам обработки относятся моменты ввода ИЭ в систему, начала и завершения обработки на программном комплексе обработки, начала и окончания обработки на супервизоре и др. Каждый этап связывается с моментом его возникновения, программой, реализующей процесс, и ресурсом, обслуживающим процесс.

Можно считать, что в определенные промежутки времени входной поток обладает свойством стационарности и можно априорно, на основании статистических данных, задать количество ресурсов, необходимых для его обработки. Предлагается

в супервизоре СКОИ реализовать оптимальное распределение вычислительного ресурса системы между ПКО. Количество вычислительных ресурсов выбирается так, чтобы СКОИ могла обрабатывать входной поток за требуемое время. Производительность работы СКОИ измеряется по использованию ресурса процессорного времени и оперативной памяти.

Распределение ПКО по ВМ можно интерпретировать как задачу целочисленного нелинейного программирования. СКОИ имеет сложную внутреннюю структуру, которая отражается в нелинейных ограничениях. Сложная структура СКОИ и постановка задачи требует использования нестандартных методов решения [4, 5].

Генетические алгоритмы являются одним из перспективных методов при решении задач оптимизации с нелинейной целевой функцией и нерегулярным пространством поиска [3]. Они обладают высокой робастностью для предотвращения попадания в локальные минимумы и способны получать действительно глобальное оптимальное решение. Кроме того, это методы нулевого порядка, не использующие информацию о производных целевой функции.

Оптимальное распределение ресурсов в СКОИ

Математическая модель задачи представляется в виде:

$$x_{opt} = \min_{R\in\Omega} \left\{ S \cdot P\left(x^*\right) \right\},\ \Omega = 1,..., R^{\max}, \qquad (1)$$

$$x^* = \arg\max_{x\in X} \left\{ \alpha \sum_{i=1}^{R} \frac{1}{T_i} \left[\sum_{j=1}^{M} t_j x_{ij} \right] + (1-\alpha) \sum_{i=1}^{R} \frac{1}{V_i} \left[\sum_{j=1}^{M} v_j x_{ij} \right] \right\}, \qquad (2)$$

$$\sum_{j=1}^{M} t_j x_{ij} \le T_i,\ \sum_{j=1}^{M} v_j x_{ij} \le V_i,\ i = \overline{1, R}, \qquad (3)$$

$$\sum_{i=1}^{R} x_{ij} = C_j,\ j = \overline{1, M}, \qquad (4)$$

$$H_i\left(x\right) = 0,\ i = \overline{1, K}. \qquad (5)$$

Псевдооптимальное распределение ПКО по ВМ представляет собой матрицу $x_{opt}=\|x_{ij}\|_{R\times M}$, где R – число ВМ, M – число ПКО. X – множество возможных распределений, определяемых ограничениями (3–5).

Критерием оптимальности является минимальное число ВМ, размещение на которых заданного числа ПКО удовлетворяет ограничениям (3–5). СКОИ имеет $R^{\max}$ число ВМ, минимальное количество ВМ определяется итерационной процедурой нахождения хотя бы одной точки, удовлетворяющей ограничениям (3–5) при текущем числе ВМ $1\le R\le R^{\max}$, после этого для найденного R ищется оптимальное распределение.

В работе используется многокритериальная целевая функция для оценки производительности системы по процессорному времени и оперативной памяти. Для определения решения оптимального в смысле Парето используется метод взвешенной функции (6), где $\alpha\in(0,1)$ – параметр скаляризации.

Наличие ограничений делает невозможным использование методов поиска безусловного экстремума. Ограничения (3) определяют пределы загрузки для каждого ВМ. Вектора $T=(T_1,T_2,...,T_R)^T$, $V=(V_1,V_2,...,V_R)^T$ задают ресурс процессорного времени и оперативной памяти имеющихся ВМ. Вектора $t=(t_1,t_2,...,t_M)^T$, $v=(v_1,v_2,...,v_M)^T$ задают ресурсы процессорного времени и оперативной памяти необходимые для работы соответствующих ПКО. Ограничения (4) задают число ПКО разного типа, необходимых для обработки входного потока. Коэффициенты $C=(C_1,C_2,...,C_M)^T$ определяются на основании статистических данных в соответствии с заданными требованиями по производительности.

Внутренняя структура СКОИ отражается в нелинейных ограничениях (5). Через нелинейные ограничения задается возможность запуска различного вида ПКО на одном ВМ, учет производительности ПКО, отражение изменения производительности в зависимости от расположения ПКО в структуре конвейера. Для работы с нелинейными ограничениями применяется подход [6], использующий запись целевой функции в форме функции Лагранжа со штрафом (7). Преобразованная таким образом целевая функция является нелинейной, что делает невозможным использование методов линейного программирования. Ограничения (5) являются недиффренцируемыми, что не позволяет использовать градиентные методы оптимизации.

$$f\left(x\right) = \alpha \sum_{i=1}^{R} \frac{1}{T_i} \left[\sum_{j=1}^{M} t_j x_{ij} \right] + (1-\alpha) \sum_{i=1}^{R} \frac{1}{V_i} \left[\sum_{j=1}^{M} v_j x_{ij} \right], \qquad (6)$$

$$\Theta\left(x, \lambda, \rho\right) = f\left(x\right) - \sum_{i=1}^{K} \lambda_i H_i\left(x\right) + \frac{\rho}{2} \sum_{i=1}^{K} H_i\left(x\right)^2, \qquad (7)$$

где $f(x)$ – целевая функция; $\Theta(x,\lambda,\rho)$ – преобразованная целевая функция; λ_i – множители Лагранжа при нелинейных ограничениях; ρ – параметр штрафа.

Для определения экстремума целевой функции (2) используется генетический алгоритм. Учет нелинейных ограничений произвольного вида в преобразованной целевой функции (7) и использование генетического алгоритма для поиска экстремума позволяет существенно расширить класс моделей, применяемых для описания СКОИ.

«Жадный» алгоритм поиска

Задача построения оптимального распределения является задачей целочисленного программирования, но использование только целочисленного генетического алгоритма сравнимо по скорости работы с полным перебором всех возможных распределений. Поэтому построение распределения ПКО по ВМ производится в два этапа: с помощью генетического алгоритма находится нецелочисленный максимум по критерию (2), затем округление и преобразование полученного решения x^* для приведения к виду, удовлетворяющему ограниче-

ниям, по формуле (1). Свертка $S \cdot P$ задает округление и преобразование. Построение псевдооптимального распределения сводится к нахождению в окрестности x^* целочисленного решения x^{opt}.

Предполагается, что после округления часть из ограничений (3) и (4) удовлетворяется, и за конечное число атомарных преобразований распределение x^* можно привести к x^{opt}. Под атомарным преобразованием понимается изменение на единицу любого элемента распределения ПКО по ВМ. Преобразование P можно представить в виде поиска в пространстве состояний, где текущее распределение можно рассматривать как состояние $\|x_{ij}^q\|$, $q \geq 0$, $q \in \aleph$, округленное распределение как начальное $x^* = \|x_{ij}^0\|$, соответственно цель поиска – $x^{otp} = \|x_{ij}^Q\|$. Поскольку размерность матрицы распределения может быть большой и в задаче наблюдается большой коэффициент ветвления, то чтобы избежать комбинаторного взрыва, предлагается использовать «жадный» алгоритм поиска [7]. Использование эвристики для выбора наиболее перспективных состояний и механизма возвратов делает «жадный» алгоритм перспективным методом для решения подобных задач.

Для реализации эвристик могут быть использованы следующие стратегии поиска:

1. Удовлетворение ограничений (4): уменьшение $x_{ij} : \sum_i x_{ij} > C_j$, увеличение $x_{ij} : \sum_i x_{ij} < C_j$.
2. Использование поиска в глубину, чтобы удовлетворить ограничения (3) и (5).

Реализация

Для реализации предложенного подхода на базе Matlab R2009a был создан макет программной системы, позволяющей производить расчет псевдооптимального распределения ПКО по ВМ. Стандартных средств Matlab было недостаточно для решения поставленной задачи, поэтому при работе системы используется модифицированная версия пакета Matlab genetic algorithm and direct search toolbox [8]. Часть компонентов пакета была подвергнута рефакторингу и реинжинирингу, чтобы удовлетворять требованиям по скорости работы для задач большой размерности. Система имеет расширенный набор параметров и позволяет отображать информацию о работе алгоритма. Также присутствует возможность использовать целочисленный вариант генетического алгоритма.

Рассмотрим систему конвейерной обработки информации с 20-ю вычислительными модулями. Построим псевдооптимальное распределение 10 ПКО по вычислительным модулям системы.

Число ПКО необходимых для обработки заданного входного потока представлены в табл. 1. Данные в таблицах задаются в виде безразмерных усредненных единиц. Все вычислительные модули одного типа, соответственно ресурсы процессорного времени и оперативной памяти задается виде $T_i = V_i = 100$. Различные характеристики вычислительных модулей могут быть учтены через весовые коэффициенты в нелинейных ограничениях. Ресурсы процессорного времени и оперативной па-

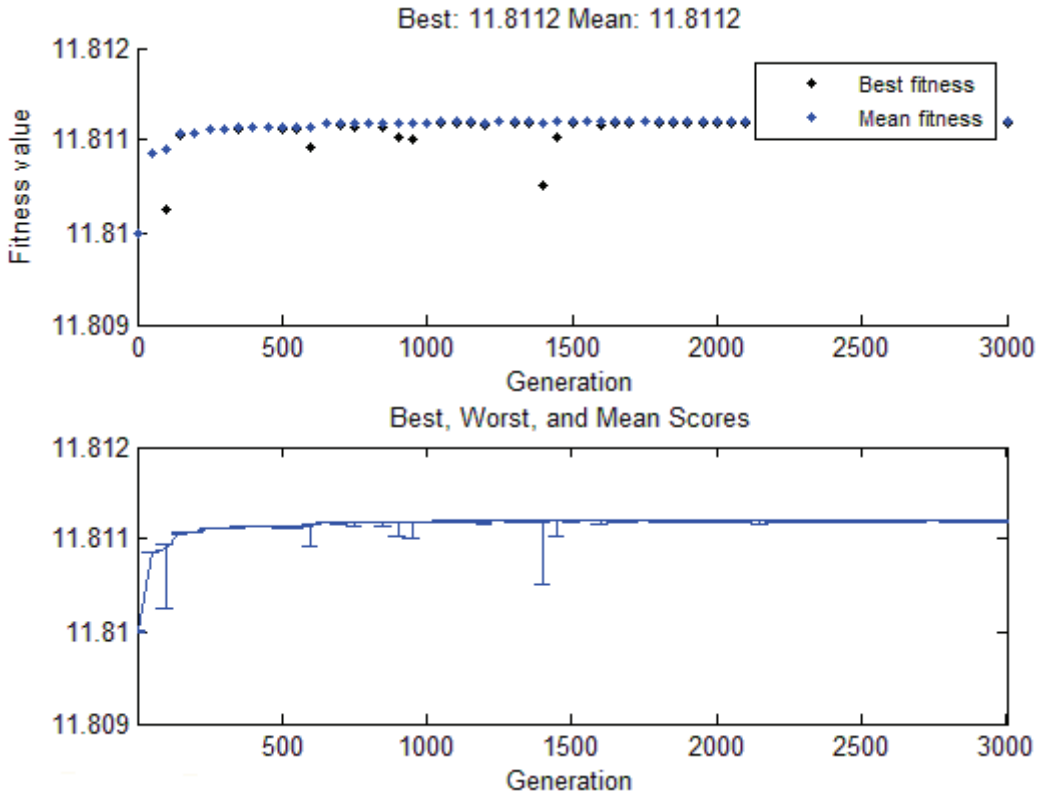

Рисунок. *Распределение по поколениям лучших, худших особей и среднее значение функции приспособленности по популяции*

мяти необходимые для работы соответствующих ПКО представлены в табл. 2.

Таблица 1. *Число ПКО разного типа, необходимых для обработки заданного входного потока*

ВМ	1	2	3	4	5	6	7	8	9	10
C	12	5	10	14	9	13	9	10	8	6

Таблица 2. *Ресурсы процессорного времени и оперативной памяти необходимые для работы ПКО*

ВМ	1	2	3	4	5	6	7	8	9	10
t	10	20	7	15	9	22	35	11	9	3
v	5	15	8	18	3	13	15	10	2	11

Расчеты показывают, что минимальное число ВМ, при котором удовлетворяются ограничения, $R^{\min}=14$. $R^{\min}$ обеспечивает максимальную загрузку каждого из используемых вычислительных модулей. Увеличение числа ВМ существенно уменьшает загруженность каждого из используемых модулей.

Таблица 3. *Псевдооптимальное распределение ПКО по ВМ*

	ПКО									
ВМ	0	1	0	1	2	1	0	2	0	1
	1	0	0	0	0	1	1	3	0	0
	1	1	0	1	0	1	0	0	3	2
	0	1	2	0	1	1	1	0	0	0
	0	0	0	0	2	0	2	1	0	0
	2	1	1	0	0	0	1	0	2	0
	0	0	1	1	1	1	1	0	1	0
	2	0	0	2	0	1	0	2	0	0
	1	0	1	1	0	1	1	1	0	0
	2	0	1	2	1	1	0	0	1	1
	0	0	1	3	0	2	0	0	0	1
	0	0	2	1	1	1	1	0	0	0
	1	0	1	1	1	1	1	0	0	0
	2	1	0	1	0	1	0	1	1	1

На рисунке приведена работа генетического алгоритма по формированию оптимального нецелочисленного распределения, показано изменение значений функций приспособленности при увеличении числа поколений. При работе алгоритма используются элитарная стратегия поиска, кроссинговер методом равномерного скрещивания, селекция методом рулетки, вероятность мутации равна 20 %.

Псевдооптимальное распределение ПКО по ВМ для данной СКОИ представлено в табл. 3. Табл. 4 содержит полученные показатели загруженности ресурсов по процессорному времени и оперативной памяти.

Таблица 4. *Показатели загруженности для каждого вычислительного модуля*

ВМ	1	2	3	4	5	6	7	8	9	10	11	12	13	14
T_Σ	100	100	100	100	99	100	97	94	100	100	99	95	98	100
V_Σ	83	63	79	62	46	52	59	79	69	83	99	65	62	79

Заключение

Предложенный подход может быть использован для нахождения псевдооптимального распределения вычислительных ресурсов в задачах большой размерности.

Такой подход является особенно актуальным при проектировании и функционировании СКОИ развернутых на площадках, инфраструктура которых требует экономии ресурсов. Также он может быть использован для определения оптимального запрашиваемого виртуального пула ресурсов при реализации облачных вычислений.

СПИСОК ЛИТЕРАТУРЫ

1. Васильев В.И., Ильясов Б.Г. Интеллектуальные системы управления. Теория и практика. – М.: Радиотехника, 2009. – 392 с.
2. Altiok T., Melamed B. Simulation modeling and analysis with Arena. – Burlington: Elsevier Inc., 2007. – 440 p.
3. Лю Б. Теория и практика неопределенного программирования / Б. Лю; Пер. с англ. – М.: БИНОМ. Лаборатория знаний, 2005. – 416 с.
4. Баранов В.И., Стечкин Б.С. Экстремальные комбинаторные задачи и их приложения. – 3-е изд., исправ. – М.: Физматлит, 2006. – 240 с.
5. Парадимитриу Х., Стайглиц К. Комбинаторная оптимизация: Алгоритмы и сложность. – М.: Мир, 1984. – 510 с.
6. Conn A., Gould N., Toint Ph. A globally convergent augmented lagrangian algorithm for optimization with general constraints and simple bounds // SIAM Journal on Numerical Analysis. – 1991. – V. 2. – № 2. – P. 545–572.
7. Люгер Дж. Искусственный интеллект: стратегии и методы решения сложных проблем, 4-е изд. – М.: Вильямс, 2003 – 864 с.
8. Genetic algorithm and direct search toolbox user's guide. – The MathWorks, Inc., 2006.

Поступила 18.10.2011 г.